

Poster Abstract

Secure Locations: Routing on Trust and Isolating Compromised Sensors in Location-Aware Sensor Networks

Sapon Tanachaiwiwat¹, Pinalkumar Dave¹, Rohan Bhindwale², Ahmed Helmy¹

1. Department of Electrical Engineering – Systems 2. Department of Computer Science
University of Southern California, Los Angeles, CA. 90089

1 (213) 740 9135

{tanachai, pdave, bhindwal, helmy} @usc.edu

ABSTRACT

In data-centric sensor networks, where data processing and transfer are oblivious to node IDs, conventional node-based security models are not suitable. We introduce the novel concept of *secure locations* to address non-cooperative and malicious behavior in location-aware sensor networks. Our architecture also introduces a scalable *trust-based routing* protocol (*TRANS*) to track, update and route around untrusted locations using variants of geographic and trajectory routing. As part of our protocol we provide an efficient algorithm for identifying and isolating misbehaving or compromised sensors based on their approximate locations. Our simulations show the efficacy of our approach in malicious node isolation and route infection reduction.

Categories and Subject Descriptors

C.2.2 [Computer-Communication Networks]: Network Protocols Routing Protocols

C.4.6 [Operating Systems]: Security and Protection Information Flow Controls

General Terms: Algorithm, Design, Security

Keywords: Secure Location, Sensor Networks, Trust Routing, Location Awareness

1. INTRODUCTION

In location-aware sensor networks, a large number of distributed sensors must collaborate to deliver the requested information to the sink(s). Such scenario assumes trust relationships between sensor nodes. However, in critical and sensitive mission such as military operations, sensors may fail or be compromised in a way that renders them malicious or (at least) non-cooperative. In this work we target a misbehavior model in which a compromised or failed sensor node consistently drops data packets while possibly participating in signaling and routing protocols. Our initial work focuses on static sensor networks in which geographic routing is used.

Our architecture is oblivious to the identity of individual sensors (which fits the data-centric model [1]) where we propose the novel concept of secure locations. We also introduce a trust routing protocol for our architecture, that aims to select trusted paths that do not include misbehaving nodes, by identifying the insecure

locations and routing around them efficiently. This approach scales well by keeping record of locations instead of sensor nodes.

Our trust routing mechanism uses the asymmetrical authentication scheme μ TESLA [2] to form the chain of trust. Security Protocols for Sensor Networks (*SPINS*) does not address the problem of compromised sensors. Our trust routing protocol also investigates the behavior of the nodes en-route by using a compact trust routing table that records the cooperativeness of locations. For insecure location discovery we propose and compare several probing techniques; expanding TTL search and one shot. We shall show how using these techniques, along with location information also helps in cheat-proofing the system, since a non-cooperative sensor cannot pretend to be at a location distant from its actual location.

This paper discusses two main concepts of our architecture: (i) trust routing and (ii) insecure location discovery and isolation. Also we present details of the relevant security mechanisms, and results from preliminary experiments.

2. TRUST ROUTING FOR LOCATION-AWARE SENSOR NETWORKS (*TRANS*)

We propose a new routing protocol to account for the non-cooperation and malicious behavior of nodes efficiently. *TRANS* uses the concept of trust to select a secure path and avoid insecure locations. We assume sensors know their (approximate) locations and that geographic routing (e.g., *GPSR* [3]) is used. We also assume that all destination nodes use the loose-time synchronization asymmetric mechanism, *TESLA*, to authenticate all requests and that the shared encryption key will be carried with the authenticated message from sink or base station to ensure message confidentiality. Based on this information, each node initializes trust values for its neighbors' locations. A trusted neighbor is a node that can decrypt the request and has enough trust value (based on forwarding history as recorded by the sink and other intermediate nodes). A sink sends a message only to its trusted neighbors for the destined location. Those neighbors correspondingly forward the packet to their trusted neighbors that have the nearest location to destination. Thus the packet reaches the destination along a path of trusted nodes.

The route selected by *TRANS* may not be optimal in terms of hop count but provides safe and unaltered delivery of data. The sensors and sinks monitor the activities of their neighbors and adjust their trust values accordingly. After excessive packet drops or on receiving compromised data the sink initiates a search for insecure locations along the path. On discovery of such location the sink

records the insecure-location. This information is later used to either route the packet through *detour points* inserted in the packet header, or to propagate a location *blacklist* that helps route around the insecure-locations using standard geographic routing.

2.1 Challenges: Specifying and dynamically adjusting the location trust parameters present an interesting research challenge in our design. If these parameters are not set carefully, cooperative location may be tagged as malicious due to their proximity to insecure location. This is called the *infection effect*, which we study in this work. Hence it is important to study mechanisms for identification and isolation of insecure location.

2.2 Assumptions: The sink is always trusted and cannot be compromised. We also assume that sensor network is dense enough for selecting new sensors to create new routes.

2.3 Details of Trust Table and related security mechanisms

Each node calculates trust values for its neighbors' location based on trust parameters and fills a trust table based on measurements, the malicious node isolation protocol, and, optionally, sharing blacklists (using the chain-of-trust concept). The sink node assigns the shared key for the entire sensor network. Only the destination authenticates the sink request and only the sink node authenticates the data from the destination for energy saving purposes.

This is a lightweight approach as compared to the watch-dog mechanism [5] because each nodes monitors its neighbors only when necessary. The trust table in each sensor contains the following items: Cryptography (*C*), Availability (*A*), Packet forwarding (*P*) and Trust value (*T*). The Trust value for location (or direction) *n*, can be calculated from the product of *C_n*, *A_n* and *P_n*. If the trust value is below a specific trust threshold, then this location is avoided when forwarding packets (e.g., by using the second nearest location, using detour points in modified GPSR or using an alternative route-around approach such as Trajectory Based Forwarding[4].

2.4 Routing Mechanism

The sink creates a message with source location, destination location, (optionally) detour locations and authentication message (MAC). It encrypts this message with its share key and broadcasts it. Only the neighbors whom the sink trusts and who know its shared key will be able to decrypt the request. The trusted neighbor decrypts the request, adds its location, encrypts the message with its share key and sends it to neighbors closer to the destination. This way the packet traverses a path of trusted neighbors using location-aware geographic routing. The destination verifies the authenticity of source node by the authentication message. After the sink's request packets are authenticated, the destination creates a reply encrypted with the shared key. The reply reaches the sink traversing a reverse path.

2.5 Identifying and Isolating Insecure Location

We propose and study several schemes to identify and isolate insecure locations. The first is Expanding TTL Search (*ETS*) where the sink marks data packets with increasing hop-count. Each intermediate node decrements the hop-count before forwarding. When hop count reaches zero at a node, that node sends ACK to the source informing it of its location and that the packet was received safely. Hence, the source identifies that part of the path as

safe and increases the hop count in subsequent packets. To reduce the search delay the destination may also participate in finding the malicious node. Alternatively the TTL can also be increased exponentially rather than linearly and may also be restricted to a small number. The sink identifies the insecure location from these responses; this scheme incurs less delay than *ETS*. The last scheme is *one shot*, where the sink node sends only one probe message along the path to which each node en-route replies with its location. This scheme reduces the number of packets sent and reduces discovery delays.

We introduce two schemes for isolating insecure locations: (a) black list flooding and (b) embedded black list (or detour points). In the first scheme the sink floods the black list to the vicinity of the insecure location. This approach does not require modification of *GPSR* routing or to the packet header because the non-cooperative node (at the insecure-location) will be simply removed from the neighbor list and will not be selected to participate in any routing activity. In the second approach the sink includes the black list information in the header of packet and sends directly to a detour point. This approach incurs less packet overhead but requires modification of packet headers and possible simple extensions to *GPSR* to route to detour points.

3. EVALUATION

We consider the following parameters for evaluation: packet delivery ratio, added overhead, goodput and route infection ratio (number of routes passing through malicious nodes over the total number of routes). The simulations show that our architecture using the *TRANS* with one-shot/black-list flooding over grid topology with single insecure location can improve the goodput of network by 40% when compared to the same scenario without trust routing. It also shows that our protocol does not incur extra message overhead if there is no insecure location in the network.

4. CONCLUSION

We proposed secure locations concept for location-aware sensor network, trust routing protocol (*TRANS*) along with two schemes for malicious location discovery (*ETS* and *one-shot*) and two schemes for isolation (Blacklist Flooding and Embedded Blacklist). Our initial results show the significant effects of route infection. Our simulation shows that our architecture can achieve node isolation, increase robustness against transient failures, and alleviate route infection effects. The main contribution lies in the explicit design trade-off between secure/trusted routing and shortest path routing, the illustration of the route infection problem and the introduction of several node isolation schemes.

5. REFERENCES

- [1] C. Intanagonwivat, R. Govindan and D. Estrin "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks" ACM MobiCOM, August 2000.
- [2] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", ACM MobiCOM, July 2001.
- [3] B. Karp, H. T. Kung "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks" ACM MobiCOM, Aug 2000.
- [4] D. Niculescu, B. Nath, "Trajectory-based forwarding and its applications" ACM MobiCOM 2003. To appear.
- [5] S. Marti, T.J. Giuli, K. Lai, M. Baker "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", ACM MobiCOM, August 2000.